Implementation of MyCalc

```
import java.awt.*;
import java.awt.event.*;
class MyCalc extends WindowAdapter implements ActionListener{
  Frame f;
Label l1;
Button b1,b2,b3,b4,b5,b6,b7,b8,b9,b0;
Button badd,bsub,bmult,bdiv,bmod,bcalc,bclr,bpts,bneg,bback;
double xd;
double num1,num2,check;

MyCalc(){
  f= new Frame("MY CALCULATOR");
// INSTANTIATING COMPONENETS
l1=new Label();
l1.setBackground(Color.LIGHT_GRAY);
l1.setBounds(50,50,260,60);



b1=new Button("1");
  b1.setBounds(50,340,50,50);
b2=new Button("2");
  b2.setBounds(120,340,50,50);
b3=new Button("3");
  b3.setBounds(190,340,50,50);
b4=new Button("4");
```

```
  b4.setBounds(50,270,50,50);
b5=new Button("5");
  b5.setBounds(120,270,50,50);
b6=new Button("6");
  b6.setBounds(190,270,50,50);
b7=new Button("7");
  b7.setBounds(50,200,50,50);
b8=new Button("8");
  b8.setBounds(120,200,50,50);
b9=new Button("9");
  b9.setBounds(190,200,50,50);
b0=new Button("0");
  b0.setBounds(120,410,50,50);
bneg=new Button("+/-");
  bneg.setBounds(50,410,50,50);
bpts=new Button(".");
  bpts.setBounds(190,410,50,50);
bback=new Button("back");
 bback.setBounds(120,130,50,50);

badd=new Button("+");
  badd.setBounds(260,340,50,50);
bsub=new Button("-");
  bsub.setBounds(260,270,50,50);
bmult=new Button("*");
  bmult.setBounds(260,200,50,50);
```

```
bdiv=new Button("/");
  bdiv.setBounds(260,130,50,50);
bmod=new Button("%");
  bmod.setBounds(190,130,50,50);
bcalc=new Button("=");
  bcalc.setBounds(245,410,65,50);
bclr=new Button("CE");
  bclr.setBounds(50,130,65,50);


b1.addActionListener(this);
b2.addActionListener(this);
b3.addActionListener(this);
b4.addActionListener(this);
b5.addActionListener(this);
b6.addActionListener(this);
b7.addActionListener(this);
b8.addActionListener(this);
b9.addActionListener(this);
b0.addActionListener(this);

bpts.addActionListener(this);
bneg.addActionListener(this);
bback.addActionListener(this);

badd.addActionListener(this);
```

```
bsub.addActionListener(this);
bmult.addActionListener(this);
bdiv.addActionListener(this);
bmod.addActionListener(this);
bcalc.addActionListener(this);
bclr.addActionListener(this);

f.addWindowListener(this);
//ADDING TO FRAME
f.add(l1);
f.add(b1); f.add(b2); f.add(b3); f.add(b4); f.add(b5);f.add(b6); f.add(b7);
f.add(b8);f.add(b9);f.add(b0);

f.add(badd); f.add(bsub); f.add(bmod); f.add(bmult); f.add(bdiv);
f.add(bmod);f.add(bcalc);

f.add(bclr); f.add(bpts);f.add(bneg); f.add(bback);

f.setSize(360,500);
f.setLayout(null);
f.setVisible(true);
}
            //FOR CLOSING THE WINDOW
public void windowClosing(WindowEvent e) {
  f.dispose();
}
```

```
public void actionPerformed(ActionEvent e){
  String z,zt;
                    //NUMBER BUTTON
if(e.getSource()==b1){
 zt=l1.getText();
  z=zt+"1";
  l1.setText(z);
}
if(e.getSource()==b2){
zt=l1.getText();
z=zt+"2";
l1.setText(z);
}
if(e.getSource()==b3){
  zt=l1.getText();
  z=zt+"3";
  l1.setText(z);
}
if(e.getSource()==b4){
  zt=l1.getText();
  z=zt+"4";
  l1.setText(z);
}
if(e.getSource()==b5){
  zt=l1.getText();
  z=zt+"5";
```

```
  l1.setText(z);
}
if(e.getSource()==b6){
  zt=l1.getText();
  z=zt+"6";
  l1.setText(z);
}
if(e.getSource()==b7){
  zt=l1.getText();
  z=zt+"7";
  l1.setText(z);
}
if(e.getSource()==b8){
  zt=l1.getText();
  z=zt+"8";
  l1.setText(z);
}
if(e.getSource()==b9){
  zt=l1.getText();
  z=zt+"9";
  l1.setText(z);
}
if(e.getSource()==b0){
  zt=l1.getText();
  z=zt+"0";
  l1.setText(z);
```

```
}

if(e.getSource()==bpts){  //ADD DECIMAL PTS
  zt=l1.getText();
  z=zt+".";
  l1.setText(z);
}
if(e.getSource()==bneg){ //FOR NEGATIVE
  zt=l1.getText();
  z="-"+zt;
  l1.setText(z);
}

if(e.getSource()==bback){  // FOR  BACKSPACE
  zt=l1.getText();
  try{
    z=zt.substring(0, zt.length()-1);
    }catch(StringIndexOutOfBoundsException f){return;}
  l1.setText(z);
}
              //AIRTHMETIC BUTTON
if(e.getSource()==badd){                  //FOR ADDITION
  try{
    num1=Double.parseDouble(l1.getText());
    }catch(NumberFormatException f){
      l1.setText("Invalid Format");
```

```
    return;
   }
  z="";
  l1.setText(z);
  check=1;
}
if(e.getSource()==bsub){                //FOR SUBTRACTION
  try{
   num1=Double.parseDouble(l1.getText());
   }catch(NumberFormatException f){
    l1.setText("Invalid Format");
    return;
   }
  z="";
  l1.setText(z);
  check=2;
}
if(e.getSource()==bmult){               //FOR MULTIPLICATION
  try{
   num1=Double.parseDouble(l1.getText());
   }catch(NumberFormatException f){
    l1.setText("Invalid Format");
    return;
   }
  z="";
  l1.setText(z);
```

```
  check=3;
}
if(e.getSource()==bdiv){                //FOR DIVISION
  try{
    num1=Double.parseDouble(l1.getText());
    }catch(NumberFormatException f){
      l1.setText("Invalid Format");
      return;
    }
  z="";
  l1.setText(z);
  check=4;
}
if(e.getSource()==bmod){                //FOR MOD/REMAINDER
  try{
    num1=Double.parseDouble(l1.getText());
    }catch(NumberFormatException f){
      l1.setText("Invalid Format");
      return;
    }
  z="";
  l1.setText(z);
  check=5;
}
                    //RESULT BUTTON
if(e.getSource()==bcalc){
```

```
  try{
    num2=Double.parseDouble(l1.getText());
    }catch(Exception f){
      l1.setText("ENTER NUMBER FIRST ");
      return;
    }
  if(check==1)
    xd =num1+num2;
  if(check==2)
    xd =num1-num2;
  if(check==3)
    xd =num1*num2;
  if(check==4)
    xd =num1/num2;
  if(check==5)
    xd =num1%num2;
  l1.setText(String.valueOf(xd));
}
                    //FOR CLEARING THE LABEL and Memory
if(e.getSource()==bclr){
  num1=0;
  num2=0;
  check=0;
  xd=0;
   z="";
   l1.setText(z);
```

```
  }


}
//MAIN METHOD where objects of MyCalc is instantaiated
 public static void main(String args[]){
      new MyCalc();
  }
}
```

output:-